New deep learning models and perspectives for continuous-time glucose monitoring

Antonio Álvarez López

Departamento de Matemáticas,
Universidad Autónoma de Madrid

July 14, 2025

TED2021-131390B-I00/AEI/10.13039/501100011033

Joint work with...

Marcos Matabuena (Harvard University)

Motivation: Digital health

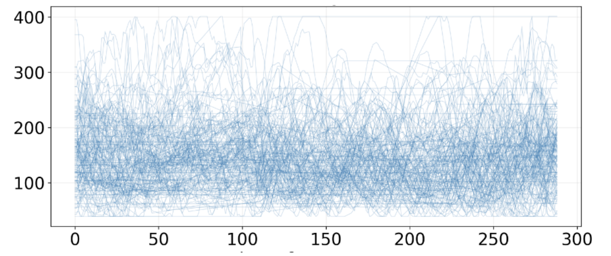
- **Continuous monitoring of glucose** in interstitial fluid without frequent finger pricks.
- **Components:**
  - Subcutaneous sensor
  - Wireless transmitter
  - Receiver or mobile app
- **Operation:** Measures every 5 minutes, displaying current levels and trends.
- **Advantages:**
  - Alerts for hypo- and hyperglycaemia
  - Analysis of curves and patterns
  - Fewer finger-stick tests
- **Limitations:** Periodic sensor replacement, higher cost.
- **Clinical use:** Mainly for type 1 diabetes and type 2 diabetes on intensive insulin therapy.

DATA

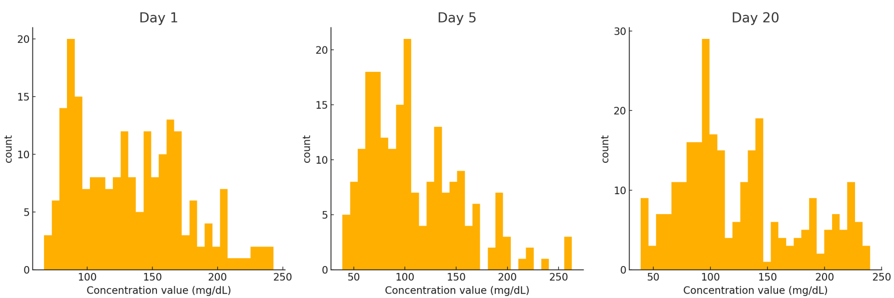
Continuous Glucose Monitoring produces data streams $(X_i)_{t_i=0}^{m}$

	0	5	10	15	20	25	30	35	40	45	50	55	60	65
1	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
2	216	222	226	246	250	254	256	250	248	246	242	238	234	230
3	114	114	112	112	110	108	108	108	108	106	104	104	102	100
4	178	180	182	184	186	188	188	190	190	190	192	192	192	186
5	178	174	170	168	168	168	168	168	166	166	164	162	164	166
6	196	196	190	186	190	194	196	196	192	190	186	186	188	190
7	70	74	76	76	76	74	72	74	74	76	78	78	78	78
8	186	186	186	188	188	190	190	192	194	196	198	200	202	204
9	222	220	216	214	212	210	210	210	206	202	202	204	208	210

$\longrightarrow$



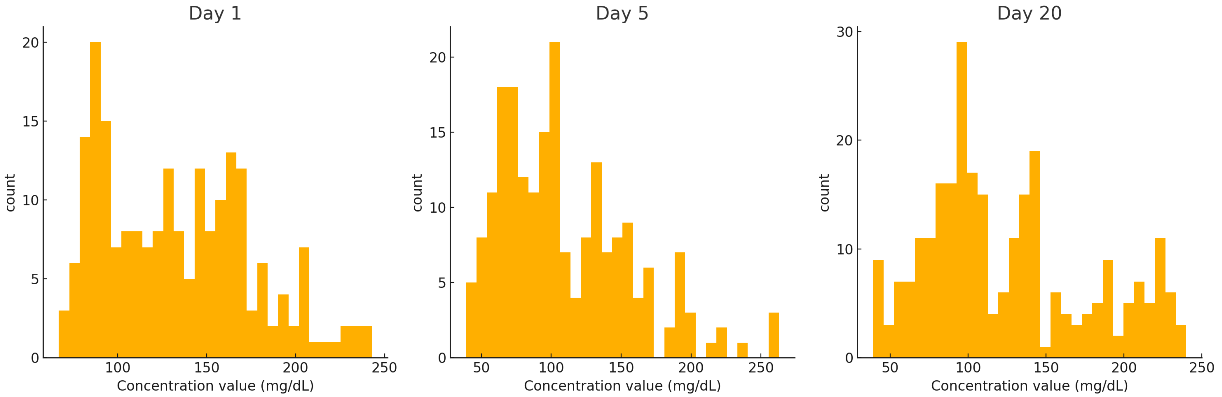
Concentration (mg/dL) vs. Measurement (total 288 per day)



This data is used to build **glucodensities** (Matabuena, Petersen, Vidal, Gude (2021)).

Goals

- **Mathematical:** Model an evolving probability distribution from random samples observed at discrete times (a time series).
- **Clinical**: Track changes in glucose distributions reflecting disease progression or treatment efficacy.



- **Difficulties:**
  - Empirical/discrete distribution
  - Multi-modal distribution (several peaks) because it mixes different times of the day $\Rightarrow$ Different metabolic states

Model overview

Gaussian mixture with dynamic weights

$$f_\theta(x,t) = \sum_{s=1}^{K} \alpha_s(t)\, \mathcal{N}(x \mid \mu_s, \Sigma_s), \qquad x \in \mathbb{R}^d$$

where[1]

$$\mu_s \in \mathbb{R}^d, \qquad \Sigma_s \in \mathscr{S}_d^+(\mathbb{R}), \qquad (\alpha_1, \dots, \alpha_K) : [0,T] \longrightarrow \Delta_{K-1} := \left\{ \alpha \in \mathbb{R}^K \;\middle|\; \alpha_i \geq 0, \sum_i \alpha_i = 1 \right\}$$

- **Interpretability:** Fixed Gaussian components (m_s, Σ_s), interpretable weights.
- By choosing K large enough, the parametric family of f_θ offers universal approximation[2] over

$$\left\{ f : \mathbb{R}^d \times [0,T] \to \mathbb{R}_{\geq 0} \;\middle|\; \int_{\mathbb{R}^d} f = 1 \right\}, \qquad \|f\| := \sup_{t \in [0,T]} \|f(\cdot,t)\|_{L^1(\mathbb{R}^d)}.$$

- See example (link)

[1] $\mathcal{N}(x \mid \mu_s, \Sigma_s) = (2\pi)^{-d/2} (\det \Sigma_s)^{-1/2} \exp\left(-\frac{1}{2}(x-\mu_s)^\top \Sigma_s^{-1}(x-\mu_s)\right)$

[2] Universal approximation of neural ODEs for dynamic behavior + Norbert Wiener. Tauberian theorems. Annals of Mathematics, 33(1):1–100, 1932.

Two-stage algorithm

1: **Input:** Time series $\{(t_i, X_i)\}_{i=1}^n$, number of Gaussians K
2: Construct aggregated data:

$$X = \bigcup_{i=1}^{n} X_i$$

3: **Initialization:** Run KMeans on X to obtain initial parameters $\{\alpha_s, \mu_s, \sigma_s^2\}_{s=1}^K$
4: **for** $\ell = 1$ to n_{iter} **do**
5: **Global GMM Fitting (Gradient Descent):** On n_{grad} iterations, find

$$\{\mu_s, \sigma_s^2\}_{s=1}^K = \arg\min_{\mu,\sigma^2} \text{MMD}^2\Big(P_X, \sum_{s=1}^K \alpha_s \mathcal{N}(\mu_s, \sigma_s^2)\Big)$$

6: **Local Weight Estimation:** For each t_i, compute

$$(\alpha_1^{(i)}, \dots, \alpha_K^{(i)}) = \arg\min_{\alpha} \text{MMD}^2\Big(P_{X_i}, \sum_{s=1}^K \alpha_s \mathcal{N}(\mu_s^*, \sigma_s^{2*})\Big)$$

7: **end for**
8: **Neural ODE Modeling:** Define the weight dynamics

$$\frac{d\alpha(t)}{dt} = f\big(\alpha(t), \psi\big), \qquad \alpha = (\alpha_1, \dots, \alpha_K).$$

9: **Parameter Estimation:** Find

$$\psi^* = \arg\min_{\psi} \sum_{i=1}^n \Big\|\alpha^{(i)} - \alpha(t_i; \psi)\Big\|^2,$$

where $\alpha(t_i; \psi)$ is the solution of the Neural ODE at time t_i.

Why two stages? Joint problem is strongly non-convex ⇒
Single pass converges to poor local minima.
Alternating strategy yields stable updates

3.–7. Discrete-time fit:

- Minimize a discrepancy (MMD^2) for each t_i.
- Yields preliminary weights $\alpha^{(i)}$.

8.–9. Continuous-time smoothing:

- Fit neural ODE to enforce temporal smoothness.
- Interpolate fitted and evolved weights.

Stage 1. Discrete-time fitting: Maximum Mean Discrepancy

Kernel $k\colon \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ symmetric, positive-definite.[3] Most common choice is the Gaussian kernel[4]

$$k(x,x') = \exp\Big(-\tfrac{\|x-x'\|^2}{2\sigma^2}\Big), \qquad \sigma = \text{median}\big\{\|x_i - x_j\|\big\}_{1\le i<j\le n}$$

Definition. Let μ, ν probability measures on $\mathbb{R}^d$. Define

$$\text{MMD}^2(\mu,\nu) = \iint_{\mathbb{R}^d\times\mathbb{R}^d} k(x,x')\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(x') + \iint_{\mathbb{R}^d\times\mathbb{R}^d} k(y,y')\,\mathrm{d}\nu(y)\,\mathrm{d}\nu(y') - 2\iint_{\mathbb{R}^d\times\mathbb{R}^d} k(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\nu(y).$$

MMD is a **distance** if and only if k is **characteristic**[5], since then $\quad \text{MMD} = 0 \iff \mu = \nu.$

Optimization advantages.

- *Efficient:* For μ sum of Gaussians and ν discrete, we derive a closed-form expression for MMD^2.
- *Robust* to the presence of outliers. Also to non-overlapping supports of μ and ν
- *Differentiable:* gradients back-propagate through k.

---

[3] I.e. for any finite set $\{x_i\}_{i=1}^n \subset \mathcal{X}$ and any real coefficients $\{c_i\}_{i=1}^n$ it holds $\sum_{i=1}^n \sum_{j=1}^n c_i\, c_j\, k(x_i, x_j) \;\geq\; 0$

[4] Heuristic: pick σ as the median of the pairwise distances in the data, so half the distances are below σ and half above.

[5] $\mu \mapsto \mathbb{E}_\mu[k(x,\cdot)]$ injective. For instance, the Gaussian kernel.

Stage 2. Weight evolution: Neural ODEs

- **Continuous-depth model.** Replace discrete network layers with an ODE:

$$\dot{\alpha}(t) = f_\phi\big(\alpha(t), t\big), \qquad \alpha(0) = \alpha_0 \quad \to \quad \text{Output: } \alpha(t) = \mathsf{ODESolve}(\alpha_0, t_0, t, f_\phi)$$

- Projection to simplex at every time:

$$\alpha(t) \leftarrow \alpha(t)/1^\top \alpha(t)$$

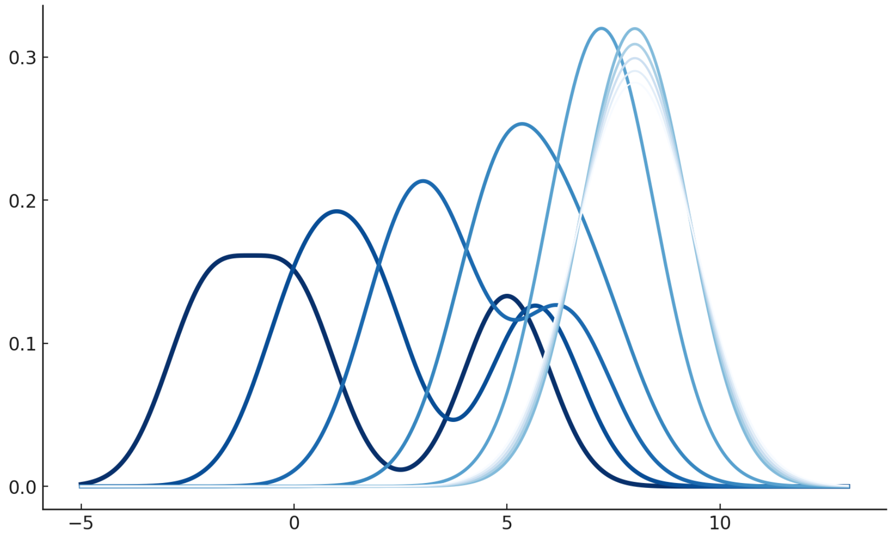
- **Training loss.**

$$\mathcal{L}_{\text{NODE}}(\phi) = \sum_i \big\|\alpha(t_i;\phi) - \alpha^{(i)}\big\|^2 + \nu\|\phi\|^2, \qquad \nu \geq 0 \text{ fixed}$$

$\nabla_\phi \mathcal{L}_{\text{NODE}}$ computed by adjoint method $\Rightarrow$ constant-memory back-propagation.

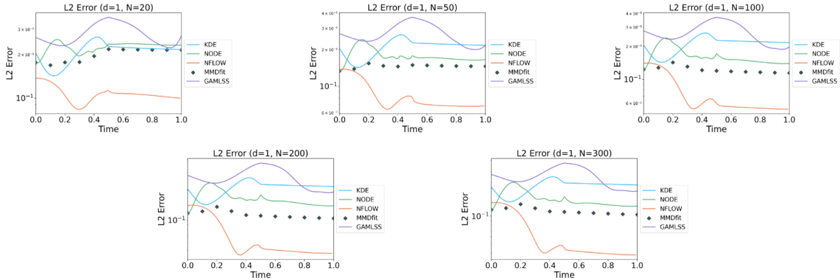
- **Key advantages**
  1. Adaptive compute matching local dynamics.
  2. Parameter-efficient (one vector field = arbitrary depth).
  3. Invertible flow (useful for generative models).
  4. Smooth latent trajectories.

Comparison with other models

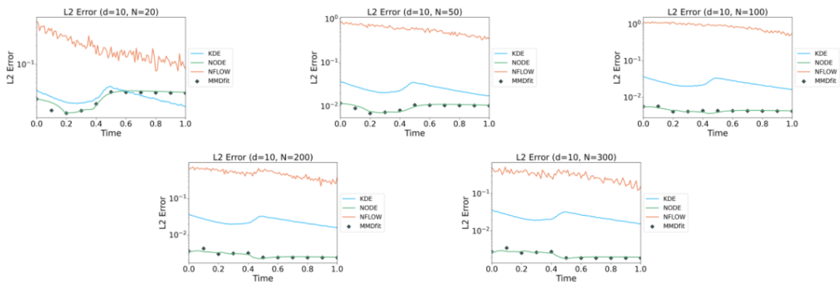


Example considered:
Normalized sum of three Gaussians drifting at constant velocity with linearly increasing variance

Comparison with other models



$d = 1$



$d = 10$

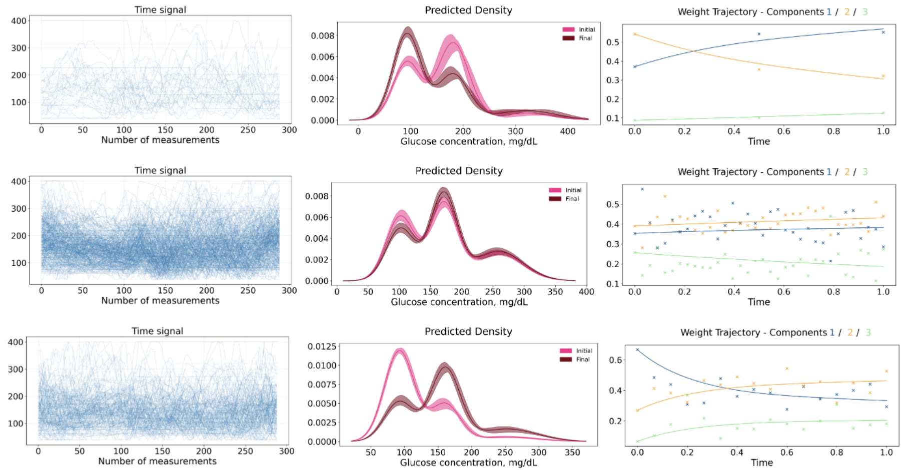


Figure 2: Each row corresponds to one of three representative patients (IDs 13, 62, and 377). Columns, left to right, show: the patient's complete measurement time series; the first (pink) and last (dark red) fitted densities with bootstrap confidence bands; the three weight trajectories $\alpha_s(t)$.

Clinical case study

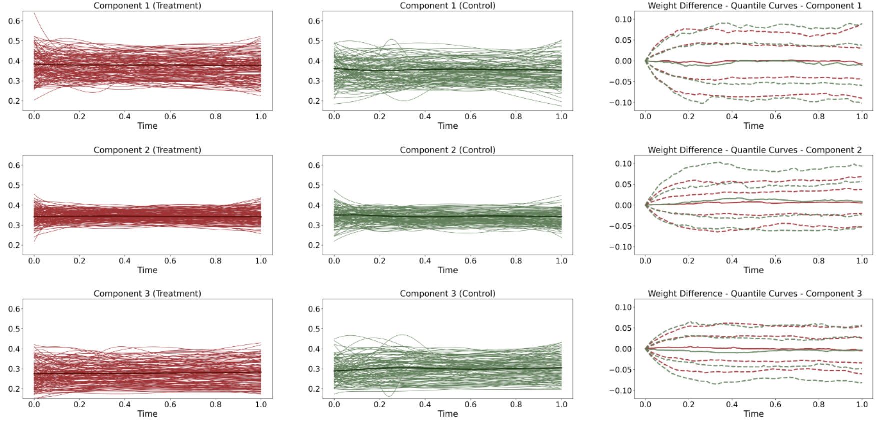


Figure 3: Rows (from top to bottom) correspond to components $s = 1, 2, 3$. Columns 1 and 2 compare all mixture weights $\alpha_s(t)$ in the treatment group (left) and the control group (right), showing individual $\alpha_s(t)$ trajectories (thin lines) and their group average (thick line). Column 3 shows, for each component, the trajectories followed by the $(0.1, 0.25, 0.5, 0.75, 0.9)$-quantiles of the process $Z_{is} = \alpha_{is}(t) - \alpha_{is}(0)$, $s = 1, 2, 3$, for each of the two groups.

(Hyper)parameters

MMD Fitting Parameters			
Parameter	Value	Parameter	Value
Max iterations	20	*Rel. tol. param.*	10^{-9}
Rel. tol. err.	10^{-7}	*Abs. tol.*	10^{-6}
Components (K)	10	*Learning rate*	0.01
Grad. steps	10	*Ridge reg.* (λ)	0.1
Neural ODE Training Parameters			
Parameter	Value	Parameter	Value
Seed	42	*Hidden dim*	200
Activation (σ)	ReLU	*Layers*	2
Optimizer	Adam	*Integration time* (T)	1.0
Step size (dt)	0.01	*Integrator*	RK4
Learning rate	10^{-3}	*L2 reg.* (λ)	0.001
Max epochs	2000	*MC samples* (n_{MC})	10000
Rel. tol.	10^{-6}	*Abs. tol.*	10^{-6}

New perspectives

- Use the whole probability distribution instead of a finite number of samples.
- Generalize to pure functional data (each sample is an element of a Hilbert space). This is motivated by biomechanics.
- Use the model to predict clinical features/outcomes by regression.
- Design new control systems driven by CGM insulin pumps based on neural ODEs.

Other Proposals

KL Divergence: Brief overview

- **Definition**: Given $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$ with $\mu \ll \nu$,

$$\mathsf{KL}(\mu||\nu) := \mathbb{E}_\mu\left[\log\left(\frac{d\mu}{d\nu}\right)\right] = \int_{\mathbb{R}^d} \log\left(\frac{d\mu}{d\nu}\right) d\mu \quad = \int_{\mathbb{R}^d} \log\left(\frac{d\mu/dx}{d\nu/dx}\right) \frac{d\mu}{dx} dx \quad \text{(if abs. cont)}.$$

- **Interpretation**: Expected excess surprise when using ν as a model for the real distribution μ.
- **Properties**:
  - **Non-symmetric**: $\mathsf{KL}(\mu||\nu) \neq \mathsf{KL}(\nu||\mu)$ (in general).
  - **Non-negative**[6]: $\mathsf{KL}(\mu||\nu) \geq 0 \quad \forall \mu, \nu$ and $\mathsf{KL}(\mu||\nu) = 0 \iff \mu = \nu$
  - **Jointly convex**:
    For any $(\mu_1, \nu_1), (\mu_2, \nu_2) \in \mathcal{P}(\mathbb{R}^d)$ and $\lambda \in [0, 1]$,

$$\mathsf{KL}\big(\lambda\mu_1 + (1-\lambda)\mu_2 || \lambda\nu_1 + (1-\lambda)\nu_2\big) \leq \lambda \mathsf{KL}\big(\mu_1||\nu_1\big) + (1-\lambda)\mathsf{KL}\big(\mu_2||\nu_2\big).$$

[6] Proof:

$$\mathsf{KL}(\mu||\nu) = \underbrace{\mathbb{E}_\mu\left[-\log\left(\frac{d\nu}{d\mu}\right)\right]}_{\text{convex}} \overset{\text{(Jensen ineq.)}}{\geq} -\log\left(\mathbb{E}_\mu\left[\frac{d\nu}{d\mu}\right]\right) = \log 1 = 0$$

BLOW-UP

- if there exists $A \subset \mathbb{R}^d$ with $\mu(A) > 0$ and $\nu(A) = 0$ then

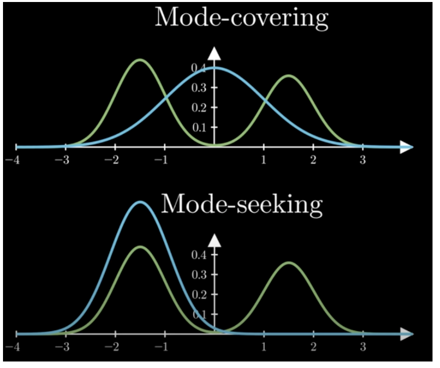
$$\mathsf{KL}(\mu \| \nu) = +\infty,$$

because $\frac{d\mu}{d\nu}(x) = +\infty$ on A. This means minimizing KL induces **mode-covering** behavior.

- if there exists $A \subset \mathbb{R}^d$ with $\nu(A) > 0$ and $\mu(A) = 0$ then

$$\mathsf{KL}(\nu \| \mu) = +\infty,$$

because $\frac{d\nu}{d\mu}(x) = +\infty$ on A. This means minimizing reverse KL induces **mode-seeking** behavior.



Experiments

$$
\begin{aligned}
\textbf{Target}: &\quad p(x) = \frac{1}{K}\sum_{k=1}^{K} \mathcal{N}(x;\, \mu_k, \Sigma_k) \\
\textbf{Model}: &\quad q_\theta(x) = \sum_{j=1}^{M} \alpha_j \, \mathcal{N}(x;\, m_j, S_j), \quad \alpha \in \Delta^{M-1} \\
\textbf{Objective}: &\quad \min_\theta \; D\big(p \,\|\, q_\theta\big), \quad D \in \{\text{KL},\ \text{reverse KL},\ L_2^2,\ \text{mix}\} \\
\textbf{Solver}: &\quad \theta_{t+1} = \theta_t - \eta_t \nabla_\theta D \quad (\text{Adam}), \\
&\quad \alpha \text{ renormalized},\ S_j \succ 0 \text{ enforced.} \\
\textbf{Outcome}: &\quad q_\theta \text{ fits } p \text{ under the chosen divergence.}
\end{aligned}
$$

- See results (link)

Possible Applications

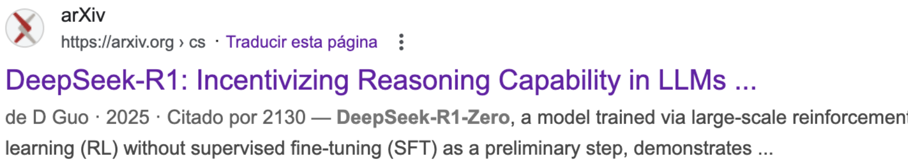
- **Source identification / localization**: KL is excellent for problems where it is critical to not "lose" any source:
  - Pollution hotspots
  - Seismic source mapping
  - Astronomical clustering
- **Risk-sensitive domains**: finance, medicine—costly to underestimate extreme events
- **Unbalanced mass**: If $\mu(\mathbb{R}^d) = \alpha \neq \beta = \nu(\mathbb{R}^d)$,

$$\mathsf{KL}(\mu||\nu) = \underbrace{\alpha \log\left(\frac{\alpha}{\beta}\right)}_{\text{Mass mismatch}} + \underbrace{\alpha\,\mathsf{KL}(\tilde{\mu}\,\|\tilde{\nu})}_{\text{Shape mismatch}} \qquad \text{with} \quad d\tilde{\mu} = d\mu/\alpha, \quad d\tilde{\nu} = d\nu/\beta.$$

People often use the extension:

$$\widetilde{\mathsf{KL}}(\mu||\nu) = \mathsf{KL}(\mu||\nu) - \alpha + \beta \qquad (\widetilde{\mathsf{KL}} = \mathsf{KL} \quad \text{if} \quad \alpha = \beta)$$

Mixture of experts (MoE)



Transformer block (token $x_i \in \mathbb{R}^d$): $\begin{cases} y_i & = \text{Normalize}\big(x_i + \text{Attention}(x_i; x_1, \ldots, x_n)\big) \\ x_{i+1} & = \text{Normalize}\big(y_i + \text{MLP}(y_i)\big) \end{cases}$

Replace the single MLP with a MoE: an ensemble of smaller, region-specialized MLPs in which only a relevant subset is activated for each input.

Mixture of experts (MoE)

$$\begin{cases} x &= y + \sum_{j=1}^{n_s} \mathrm{MLP}_j^{(s)}(y) + \sum_{j=1}^{n_r} g_j\, \mathrm{MLP}_j^{(r)}(y) \\ g_j &= \dfrac{g'_j}{\displaystyle\sum_{\ell=1}^{n_r} g'_\ell} \\ g'_j &= \begin{cases} s_j, & \text{if } s_j + b_j \in \mathrm{Top}_k\big(\{\, s_\ell + b_\ell \,\}_{\ell \in [n_r]}\big) \\ 0, & \text{otherwise,} \end{cases} \\ s_\ell &= \mathrm{Sigmoid}\big(\langle r_\ell,\, y\rangle\big) \end{cases}$$

where $n_s \geq 1$ (shared experts); $n_r \geq 1$ (routed experts); $1 \leq k \leq n_r$ (granularity: number of activated routed experts); s_j token-to-expert affinity; $r_j \in \mathbb{R}^d$ and $b_j \in \mathbb{R}$ are the centroid vector and bias of the j-th routed expert; and

$$\mathrm{Top}_k(\{u_1, \ldots, u_n\}) \quad \equiv \quad k \text{ highest values among } u_1, \ldots, u_n \in \mathbb{R}.$$

Mixture of experts (MoE)

$$\begin{cases} x &= y + \sum_{j=1}^{n_s} \mathrm{MLP}_j^{(s)}(y) + \sum_{j=1}^{n_r} g_j \, \mathrm{MLP}_j^{(r)}(y) \\ g_j &= \frac{g'_j}{\sum\limits_{\ell=1}^{n_r} g'_\ell} \\ g'_j &= \begin{cases} s_j, & \text{if } s_j + b_j \in \mathrm{Top}_k\big(\{\, s_\ell + b_\ell \,\}_{\ell \in [n_r]}\big) \\ 0, & \text{otherwise,} \end{cases} \\ s_\ell &= \mathrm{Sigmoid}\big(\langle r_\ell, y \rangle\big) \end{cases}$$

Questions

1. Understand the geometry
2. Understand the difference in dynamics with respect to the single MLP

Thanks for the attention!

Extra I: RKHS, KDE and MMD

1. Reproducing Kernel Hilbert Space (RKHS): A Hilbert space of functions where the evaluation function is linear and bounded.
This defines a positive-definite kernel with the reproducing property:

$$f(x) = \langle f,\, k(x,\cdot)\rangle_{\mathcal{H}_k}, \qquad \mathcal{H}_k := \overline{\mathrm{span}}\{k(x,\cdot)\}_{x\in\mathcal{X}}.$$

Functions in $\mathcal{H}_k$ can thus be evaluated via dot products $\Rightarrow$ kernel tricks.

2. Kernel Mean Embedding (KME): Image of a disribution p in the RKHS (it codifies all the means of functions in $\mathcal{H}_k$)

$$\mu_P = \mathbb{E}_{x\sim P}\big[k(x,\cdot)\big] \in \mathcal{H}_k, \qquad \hat{\mu}_P = \frac{1}{m}\sum_{i=1}^{m} k(x_i,\cdot).$$

μ_P represents the entire distribution; if the kernel is characteristic, the map $P \mapsto \mu_P$ is injective.

3. Maximum Mean Discrepancy (MMD): Distance between two KMEs

$$\mathrm{MMD}_k(P,Q) = \|\mu_P - \mu_Q\|_{\mathcal{H}_k}, \qquad \mathrm{MMD}_k(P,Q) = 0 \iff P = Q \quad (\text{for characteristic } k).$$

Kernel $\Rightarrow$ RKHS $\Rightarrow$ KME (distribution as a mean) $\Rightarrow$ MMD (distance between distributions)

EXTRA II: COMPUTE MMD

Closed-form expression for the MMD between a Gaussian mixture and an empirical distribution:

$$f_i(x) = \sum_{s=1}^{K} w_s \mathcal{N}(x \mid m_s, \Sigma_s).$$

At each $t_i \in \tau$, we use a Gaussian kernel k_i such as (2), with $\sigma_i^2 \approx (\operatorname{median}_{j \neq k} \|X_{t_i,j} - X_{t_i,k}\|)^2$. Thus,

$$\mathrm{MMD}^2(P_{t_i}, Q) = \sum_{s=1}^{K} \sum_{r=1}^{K} w_s w_r \, I_{i,s,r} - \frac{2}{n_i} \sum_{s=1}^{K} \sum_{j=1}^{n_i} w_s \, J_{i,s,j} + \frac{1}{n_i^2} \sum_{j=1}^{n_i} \sum_{\ell=1}^{n_i} k_i(X_{t_i,j}, X_{t_i,\ell}).$$

The two first terms admit closed-form expressions:

$$I_{i,s,r} = \frac{(\sigma_i^2)^{d/2}}{\sqrt{\det(\Sigma_s + \Sigma_r + \sigma_i^2 \mathsf{Id})}} \exp\Big(-\frac{1}{2}(m_s - m_r)^\top (\Sigma_s + \Sigma_r + \sigma_i^2 \mathsf{Id})^{-1} (m_s - m_r)\Big),$$

$$J_{i,s,j} = \frac{(\sigma_i^2)^{d/2}}{\sqrt{\det(\Sigma_s + \sigma_i^2 \mathsf{Id})}} \exp\Big(-\frac{1}{2}(X_{t_i,j} - m_s)^\top (\Sigma_s + \sigma_i^2 \mathsf{Id})^{-1} (X_{t_i,j} - m_s)\Big),$$